AD-A106 027 STANFORD UNIV CA SYSTEMS OPTIMIZATION LAB F/G 12/1

A FURTHER INVESTIGATION OF EFFICIENT HEURISTIC PROCEDURES FOR I--ETC(U)

AUG 81 F S HILLIER N00014-76-C-0418

UNCLASSIFIED SOL-81-11 NL

1 OF 1

AD A
106027

END

DATE FILMED

11-81

DTIC

LEVEL

0

Systems
Optimization
Laboratory

AD A106027

DTIC FILE COPY

DTIC
ELECTE
OCT 22 1981
S D
A

Department of Operations Research
Stanford University
Stanford, CA 94305

81 10 6 267

SYSTEMS OPTIMIZATION LABORATORY
DEPARTMENT OF OPERATIONS RESEARCH
STANFORD UNIVERSITY
STANFORD, CALIFORNIA 94305

14 81-11

A FURTHER INVESTIGATION OF EFFICIENT HEURISTIC PROCEDURES FOR INTEGER LINEAR PROGRAMMING WITH AN INTERIOR

by

Frederick S. Hillier

TECHNICAL REPORT SOL 81-11

August 1981

12 66

15

Research and reproduction of this report were partially supported by the National Science Foundation Grant MCS76-81259. This report was also issued as Report Number 100 under the Office of Naval Research Contract N00014-76-C-0418, at Stanford University.

[JNS, MCS76-81259]

408765

1. Introduction

The author [9] previously developed and evaluated some heuristic procedures for seeking a good approximate solution of any pure integer linear programming problem [illegible],

$$\text{maximize} \quad x_o = \sum_{j=1}^{n} c_j x_j \ ,$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad (i = 1, 2, \ldots, m)$$

$$x_j \text{ is a nonnegative integer} \qquad (j = 1, 2, \ldots, n) \ ,$$

having no implicit or explicit equality constraints. It was found that the procedures are extremely efficient, being computationally feasible for problems having hundreds of variables and constraints. Furthermore, they proved to be very effective in identifying good solutions, often obtaining optimal ones. Thus, the procedures provide a way of dealing with the frequently encountered integer programming problems that are beyond the computational capability of existing algorithms. For smaller problems, they also provide an advanced start for accelerating certain primal algorithms, including the author's Bound-and-Scan algorithm ~~[7]~~ e and Faaland and Hillier's Accelerated Bound-and-Scan algorithm. ~~[4].~~ In addition, Jeroslow and Smith ~~[12]~~ have found that imbedding the first part of one of these procedures inside the iterative step of a branch-and-bound algorithm can greatly improve the latter's efficiency in locating solutions whose objective function value is within a specified percentage of that for the optimal solution.

All of the procedures use the same general three-phase approach, which can be described conceptually as follows. Phase 1 identifies a general region within which to explore for good feasible solutions by finding (1) the optimal noninteger solution by the simplex method, and (2) a second point well into (or beyond) the feasible region. Phase 2 then conducts this exploration by slowly moving along the line segment from this first point to the second while searching nearby for a feasible (integer) solution. Phase 3 attempts to move from the feasible solution obtained to a succession of better ones. The final solution obtained is the desired approximate solution. If it is crucial to increase the probability of obtaining an optimal solution, one can continue by identifying many good feasible solutions in Phase 2 and then applying Phase 3 to each of them, thereby yielding multiple final solutions from which to choose.

The previous paper [9] presents alternative methods for conducting each of these phases, thereby yielding 36 distinct overall procedures. (These procedures are labeled x-y-z to indicate that Methods x, y and z are to be used in Phases 1, 2 and 3, respectively.) A program of computational experimentation identified four types of procedures (1-2-1, 2-2-1, 1-3-1, and 2-3-1) that appear to be substantially better than the others, but this experimentation was unsuccessful in detecting significant differences among the four. Furthermore, only tentative conclusions can be drawn in comparing the four alternative criteria (A, B, C, and D) for conducting a certain test in Phase 2. Thus, 16 distinct procedures (1-2i-1, 2-2i-1, 1-3i-1, and 2-3i-1 for $i = A,B,C,D$) still await definitive comparison. Another question of this study which was only partially answered concerns the best way in which to generate multiple solutions.

The present paper has two main purposes. One is to briefly present some promising new methods for conducting each of the three phases.

This is done in the next three sections. The second objective is to address some unanswered questions mentioned above (and outlined in more detail in the last section of [9]) in the broader context of these new procedures. Thus, a comprehensive testing program has been conducted to further evaluate and compare the best of the old procedures with the new ones presented here. Procedures for generating multiple solutions also are discussed and tested. The test results and conclusions are presented in the final three sections and the appendix.

Various other investigations also have proposed heuristic algorithms for integer programming in recent years. These include Reiter and Rice [14], Echols and Cooper [2], Senju and Toyoda [16], Roth [15], Kochenberger, McCarl and Wyman [13], Toyoda [17], Balas and Martin [1], and Glover [5]. Also of particular interest here is the heuristic algorithm of Ibaraki, Ohashi, and Mine [10], which extends (with some modification) the author's original heuristic procedures [9] to mixed integer programming. (See Section 9 for a comparison of this algorithm with the procedures proposed here.) In addition, Faaland and Hillier [3] have extended the analysis and development of Phase 1 methods considerably beyond the present paper.

For the sake of brevity, the presentation here will not repeat most of the relevant material from [9], so the reader is advised to first read this earlier paper and to keep it available for reference purposes as he proceeds through the following.

2. A Multiple Solution Approach to Phase 1

It would sometimes be worthwhile to take the time to generate multiple final solutions in order to try to improve upon the initial one obtained after completing all three phases once. As mentioned above, one method of doing this, based on repeated applications of Phase 2, is presented in [9]. Another approach described below is based instead on repeated applications of Phase 1.

This approach involves generating a sequence of distinct pairs of points, $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$. Recall that the original $\underline{x}^{(1)}$ is just an optimal solution to the problem without the integer restriction, as obtained by the simplex method. Each new $\underline{x}^{(1)}$ is then obtained directly from this one by choosing an adjacent extreme point in the polyhedral set of feasible solutions for this linear programming problem. This adjacent extreme point is obtained simply by performing a single pivot from the optimal basis. The method used here for choosing the sequence of adjacent extreme points is based on the size of the simplex multipliers (i.e., reduced costs). Thus, the first pivot is performed on the nonbasic variable having the smallest (negative) simplex multiplier, the second on the one having the second smallest multiplier, etc. This process was stopped after obtaining a preassigned number of new values of $\underline{x}^{(1)}$, which was taken to be five for the computational experimentation. (Note that another alternative would be to examine all of the adjacent extreme points and then choose the five best ones according to their objective function value.)

For each $\underline{x}^{(1)}$, the chosen Phase 1 method would be used to obtain the corresponding value of $\underline{x}^{(2)}$ in the usual way. Connecting these two points provides a line segment which is completely distinct from the original one, but which still passes through a promising region for searching for a good feasible solution. Therefore, Phase 2 and then Phase 3 would be applied just as before with respect to the new $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$ in order to obtain the new final solution. Doing this in turn with each new pair of such points provides the desired multiple solutions, from which the best one would be chosen as the solution to use.

A very desirable characteristic of any method of generating multiple solutions would be that each new solution generated have a relatively

high probability of being distinct from the preceding ones while still tending to have an objective function value that is at least close to and sometimes is better than that for the initial final solution. The fact that the original method frequently merely regenerated a previous final solution was a major motivation for developing the new approach described above. It was anticipated that having the Phase 2 search proceed from a distinct new line segment rather than merely moving further down the same segment might increase the chances of obtaining a distinct final solution that was still a good one.

This new approach will be designated by inserting an R (for "Repeat") before the number of the method being used for each application of Phase 1. Similarly, the original approach will be designated by inserting an R before the number of the method being used for each application of Phase 2. For example, the labeling for using procedure 1-2A-1 to generate multiple solutions by the new and original approaches would be R1-2A-1 and 1-R2A-1, respectively.

3. New Criteria in Phase 2

Recall that Methods 2 and 3 of Phase 2 involve iteratively moving from one infeasible integer solution to another which is "less infeasible" in a certain sense (or, if such an improvement is not possible, beginning another cycle of such iterations with a new starting solution). Each such move involves changing one variable by ± 1. Four different criteria for choosing the variable to be changed were presented. Criteria A and B focus exclusively on the constraints, and thereby would seem to run some risk of leading to a feasible solution with a relatively poor value of the objective function. On the other hand, criteria C and D give considerable weight to the objective function, but in a way that sometimes

gives preference to moves that point above rather than toward the feasible region, which would seem to increase the risk of overlooking feasible solutions subsequently. This analysis would suggest that a compromise between these two types of criteria might be appropriate. One such compromise is described below.

Using the notation and terminology of [9], the new criterion defines the "improvement" from changing the value of a variable x_j as

$$p = -\Delta q + c_j' \Delta x_j ,$$

where c_j' is the normalized value of c_j and Δx_j is the change in x_j. Thus, this is just the definition used for criteria A and B $(p = -\Delta q)$ except for adding a term reflecting the effect of the change on the objective function. In the case of criterion A, where the first definition of q is used, this added term has a very natural interpretation. In particular, suppose that a lower bound b_o on an acceptable value of the objective function is introduced explicitly as a constraint, $\underline{cx} \geq b_o$, and that b_o exceeds the value x_o attained by the current solution both before and after the change in x_j. Then the resulting criterion A definition of p coincides exactly with the new definition given above. Thus, in effect, the new criterion encourages large moves toward the feasible region (as with criteria A and B) but with a modified interpretation of feasibility that particularly encourages movement toward the most attractive portion of the feasible region (as with criteria C and D).

The mechanics of applying the new criterion are to proceed through Phase 2 exactly as if criterion A or B were being used (depending on

which definition of q is adopted) except when there are no $q_j^* \leq 0$ and part (c) of Step 8 is entered, in which case merely substitute the new definition of p for each p_j to be calculated. Since the results of computational testing distinctly favor criterion A over criterion B (see Sections 5 and 6), only the first definition of q (the one used by criterion A) was used with the new criterion, and this version will be called criterion E.

It should be noted that there also are variations of this criterion that conceivably could be slightly better. For example, a factor other than one could be used for the new term in order to give a different relative weighting between the constraints and objective function, or the new term could be deleted when it is positive (effectively setting b_o equal to x_o before changing a variable), etc.

Another viewpoint is that it is really combinations of moves that are particularly important in moving expeditiously to a good feasible solution. Therefore, any move that yields an improvement should be taken immediately rather than wasting time by identifying and comparing all possible improving moves for this iteration. Other such moves that are truly worthwhile should still be available for subsequent iterations. Furthermore, when the number of variables is large, this approach may greatly reduce the time required to execute Phase 2 without significantly sacrificing effectiveness.

When the first definition of q is used, this streamlined approach will be designated as criterion S. It is applied by executing Phase 2 essentially as with criterion A (or C) except that Steps 7 and 8 are bypassed. Instead, as soon as a q_j^* such that $q_j^* < q$ is found in Step 6, the iteration is terminated immediately by setting k equal to

this j and going to Step 9. On the next iteration, Step 6 would resume from where it had left off, calculating (as necessary) $q^*_{k+1}, q^*_{k+2}, \ldots, q^*_n$, $q^*_1, \ldots, q^*_k$ in this order. If there are no q^*_j such that $q^*_j < q$ on a given iteration, then one goes to Step 10.

4. New Methods for Phase 3

A drawback of the current methods for Phase 3 is that, at each iteration, one attempts to identify a better (feasible) solution only by considering certain ways of changing either one or two variables in the current solution. It sometimes is necessary to change many variables in order to reach a better solution. However, it clearly would be computationally infeasible for problems of significant size to systematically consider all ways of changing several variables simultaneously. Therefore, what is needed are methods that will efficiently consider only promising ways of changing many variables.

As suggested by Ibaraki et al [10], one approach of this kind would be to use a search similar to that employed in Phase 2. Recall that the Phase 2 search allows making many promising variable changes in succession in an attempt to eventually reach the solution of interest, namely, a good feasible solution. Essentially the same method also could be used in Phase 3 in an attempt to eventually reach the solution of interest there, namely, a better feasible solution than the best one found thus far. Thus, one would deliberately move from the current best feasible solution out of the feasible region, and then try to move through a succession of infeasible (integer) solutions that seem to be progressing toward a better feasible solution (if one exists).

Three new methods for Phase 3 (Methods 3, 4, and 5) that are based in part on this approach are presented below. Given the current best feasible solution $\underline{x}^{(L)}$ and its objective function value $x_o^{(L)} = \underline{cx}^{(L)}$, all three methods initiate the search procedure mentioned above by introducing a new constraint, $\underline{cx} \geq b_o$, where $b_o = x_o^{(L)} + 1.$*/ This has the effect of making $\underline{x}^{(L)}$ infeasible and reducing the feasible region so that it includes only better feasible solutions (if any). Thus, the goal becomes to reach some integer solution in this reduced feasible region.

Methods 3 and 4 go through n cycles to search for a better solution. Each cycle begins by changing one variable (call it x_k) in the solution $\underline{x}^{(L)}$ by either plus one (if $c_k > 0$) or minus one (if $c_k \leq 0$). Thus, the n cycles correspond to setting $k = 1, 2, \ldots, n$ in turn. This first step in each cycle has the effect of giving a new solution which usually is substantially further away from the reduced feasible region. The remaining part of the cycle then consists of essentially applying Steps 4 through 9 of Phase 2 with this new solution as the starting point. (Criterion A always was used for doing this in the computational testing since criteria E and S had not yet been developed at this stage of the testing program.) The only changes that need to be made in Steps 4-9 are the following. First, the new objective function constraint is treated just like the other functional constraints (after making the obvious conversions in format and notation), so i effectively runs from 0 to m. Second, additional changes in x_k are not allowed so $j \neq k$ throughout. Third, having $Q = \phi$ in Step 8 terminates the cycle, in which case reset $k = k + 1$

---

*This assumes that all of the c_j are integers; otherwise, set $b_o = x_o^{(L)} + \epsilon$ where ϵ is an extremely small positive constant.

and start the next cycle (if $k \leq n$).

The entire Method 3 consists of Parts I and II of Method 1 (which obtain all possible improvements by changing only one variable at a time) plus the above procedure as a replacement for Parts III to VII (which investigate certain ways of changing two variables simultaneously). Thus, after completing Part II, set $\underline{x}^{(L)} = \underline{x}$ and go into the above search procedure. If a better feasible solution is found (i.e., $q \leq 0$ in Step 5 for some iteration of some cycle), then immediately restart Phase 3 at the beginning of Part II with this new solution. However, if the search procedure goes through all n cycles without finding a better feasible solution (perhaps on a later time through), then Phase 3 terminates with the current $\underline{x}^{(L)}$ as the desired approximate solution.

Method 4 differs from Method 3 only in that all of the original Phase 3 (Method 1) is completed before entering the above search procedure. (This holds on both the first time and subsequent times through the overall process.) Thus, Method 4 is guaranteed to do at least as well as Method 1 in obtaining a good final solution, but it may require substantially more time than Method 3 without a significant increase in effectiveness.

A serious drawback of the search procedure used by both Methods 3 and 4 is that it requires more than some multiple of mn^2 elementary operations, so that the time required grows rapidly with the size of the problem (more so than the rest of the procedure). Therefore, Method 5 modifies Method 4 by streamlining the search procedure. In particular, rather than n cycles, there is only one. Furthermore, no change is made in $\underline{x}^{(L)}$ before starting the modified Steps 4 to 9 of Phase 2.

in order to avoid thereby terminating immediately because no "improvement" is possible, Steps 7 and 8 are replaced by the following. Calculate $p_1, p_2, \ldots, p_n$ in the usual way. (Again, criterion A was used for these calculations in the testing program.) Set k equal to a value such that $p_k = \max p_i$, regardless of whether this maximum is positive or not, and then go to Step 9. Thus, at each iteration, the variable is changed which has the best effect on the "infeasibility" q, even though it may actually increase q. As a result, some drifting away from the reduced feasible set may occur before the search can (hopefully) home in on an improved feasible solution. To avoid drifting indefinitely, an upper limit is imposed on the number of iterations, which was taken to be 100 for purposes of computational testing. Another danger of eliminating the requirement that q must be strictly reduced at each iteration is that it is then possible to begin cycling, whereby the same sequence of solutions is repeated ad infinitum. One inexpensive way to lesson the risk of this occurring is to impose the restriction that a variable change cannot be reversed within a certain number of iterations (taken to be five for computational testing).

Because of its ability to drift, it is conceivable that Method 5 actually would be more effective than Methods 3 and 4 in reaching a better feasible solution that requires changes in many variables.

5. Description of Test Problems

As described previously by the author [9], some 38 test problems were used to evaluate his original heuristic procedures. Most of these were Type I and Type II problems as described in Table I, where the respective parameters are integers randomly generated (using the mixed congruential method) from a (discretized) uniform distribution over the indicated interval. The Type I problems are intended to be difficult

problems that should be particularly challenging for the heuristic procedures since there is no exploitable special structure, the coefficient matrix is completely dense, feasible solutions are relatively difficult to identify, and the variables have a wide range of values to be considered. Thus, these problems should be exceptionally effective in revealing any significant differences in the effectiveness of alternative procedures. The Type II problems are intended to be representative of the multidimensional knapsack problems (all coefficients nonnegative) frequently found in practice. See [8] for a listing of these original test problems of both types (except those larger than 30 x 30).

TABLE I

DESCRIPTION OF THE RANDOMLY GENERATED TEST PROBLEMS

Parameter	Problem Type			
	I	Ia	Ic	II
c_j	[-20,79]	Type I	Type I	[0,99]
a_{ij}	[-40,59]	except	except	[0,99]
b_i	[500,999]	$P(a_{ij} = 0) = \frac{3}{4}$	$P(c_j = 0) = \frac{3}{4}$	[1000,1999]

Eighteen of these original Type I and Type II problems were used again to test the new methods presented in the preceding sections. These consisted of the 16 problems with $m = 15$, $n = 15$ for which the standard-of-comparison procedure (1-2A-1) definitely did not obtain an optimal solution (namely, problems I-2 through I-8 and II-1,3,4,7,9,11,13,14,15) plus two larger problems for which the optimal solution is unknown (II-16, which is 30 x 30, and the 30 x 60 "Large" Type I problem, labeled I-9 here).

In addition, 40 new test problems with $m = 10$, $n = 20$ were generated at the outset. These include 20 ordinary Type I problems (labeled I-101 through I-120), plus 10 Type Ia problems (Ia-1 to Ia-10) and 10 Type Ic problems (Ic-1 to Ic-10) as described in Table I. Specifically, for Ia

problems, each a_{ij} was assigned a value of zero with probability $\frac{3}{4}$; if this event did not occur, then it was assigned a randomly generated integer (possibly zero) from the interval [-40,59] in the usual way. For Ic problems, the c_j parameters were generated in an analogous way. Since "low density" problems (particularly with respect to the a_{ij}) are commonly observed in practice, these Ia and Ic problems were intended to check on the effect of this factor. No additional Type II problems were generated since these had proven to be far less challenging (i.e., the solutions obtained tended to be far closer to being optimal) than the Type I problems for the original procedures. (None of the original Type III problems were used for the same reason.)

Finally, to avoid complete reliance on randomly generated problems, use was made of the nine IBM problems presented by Haldi [6] and reproduced by Trauth and Woolsey [18]. (None of Haldi's other test problems were used since, as reported in [9], two of the most difficult ones had presented little challenge to the heuristic procedures in previous testing.)

It should be noted that all 67 of these test problems are relatively small ones compared to the sizes that would be computationally feasible for the heuristic procedures. The two reasons for doing this were to preserve a fairly limited computer budget and to permit identifying an optimal solution with an exact algorithm for most cases. The emphasis in this testing program was on evaluating the effectiveness of the procedures, as measured by the normalized deviation from optimality (defined below). Some information also was obtained on their efficiency, but no attempt was made to test their limits of computational feasibility.

All testing was done on an IBM-360/67 computer, using FORTRAN codes. Documentation of the code for the original procedures is available in a separate report [11].

6. Evaluation of Some Original Procedures and the Phase 2 Criteria

6.1. Test Results

The experimentation program began by dealing with the old unresolved question of which basic type of procedure (from among 1-2-1, 2-2-1, 1-3-1, and 2-3-1), in conjunction with which Phase 2 criterion (A, B, C, D, plus the new E and S), is best. Thus, 24 distinct procedures were to be considered. Since these procedures (except with the new Phase 2 criteria) already had been applied (see Table II in [9]) to the 18 old test problems (except for I-9), all 24 now were individually run on just the 49 new test problems plus I-9.

On 21 of these 50 problems, the same final solution was obtained by all 24 procedures. The results for the other 29 problems are shown in Table II. For each problem, this table compares the final solution obtained by each procedure (identified partially by the footnotes to the table) with the best solution obtained by any of the 24 procedures, expressed in terms of the normalized differences in their objective function values. Specifically, if x_o^B denotes the objective function value of this best value, then the normalized deviation of a given solution with objective function value x_o from this best solution is

$$\text{normalized deviation} = (x_o^B - x_o)/\left(\sum_{j=1}^{n} c_j^2\right)^{\frac{1}{2}}$$

so that this normalized deviation is just the <u>Euclidean distance</u> between the hyperplanes,

$$\sum_{j=1}^{n} c_j x_j = x_o^B \quad \text{and} \quad \sum_{j=1}^{n} c_j x_j = x_o .$$

(When the optimal solution is known, substituting its objective function value for x_o^B in the above expression yields the <u>normalized deviation from optimality</u>, which is the quantity used in several of the subsequent tables.)

For many of the problems, the four basic types of procedures obtained the same final solution for a given Phase 2 criterion, in which case the resulting normalized deviation is listed singly in the table. When these procedures obtained different solutions, the resulting normalized deviations are listed separately with footnotes identifying the procedures involved. The average normalized deviation for each procedure is given at the bottom of the table, where this average first excludes just IBM-7 (since criteria A, B, and D failed to find any feasible solution for this problem) and next excludes both IBM-7 and IBM-8 (since the IBM-8 results were dominating the first average).

Since the results shown in Table II were inconclusive in identifying the best procedure, a supplementary experimental program involving 240 additional test problems was undertaken subsequently, as described in the appendix.

In addition to the information in Table II, various other performance data also were gathered for the 50 test problems. Since these data tend to vary with problem size, they are summarized on an average basis in Table III for just the 40 problems where $m = 10$, $n = 20$. The last three columns give a grand average over all six Phase 2 criteria for the indicated basic type of procedure, expressed as an increment over the corresponding grand average for the 1-2-1 procedure. Letting

$$\underline{x} = (1-\alpha)\underline{x}^{(1)} + \alpha\underline{x}^{(2)} ,$$

the first set of rows refers to the value of α at which a feasible solution is found during the subsequent Phase 2 search. The next set of rows refers to the number of points (excluding $\underline{x}^{(1)}$) on this line segment

TABLE II

NORMALIZED DEVIATION FROM BEST SOLUTION FOR PROBLEMS WITH DIFFERENT SOLUTIONS

m	n	Problem Type & No.	Criterion used in Phase 2 with procedures 1-2-1, 2-2-1, 1-3-1 and 2-3-1*					
			A	B	C	D	E	S
10	20	I-101	0	0	0.152	0	0.152	0.152
10	20	I-102	0.052	0.052	0.052	0	0.052	0.052 1,2,4/ 0.294 3/
10	20	I-103	0.280	0	0.280	0	0	0.280
10	20	I-104	0	0.068 1,2/ 0 3,4/	0	0.068 1,2/ 0 3,4/	0	0.068 1,2/ 0 3,4/
10	20	I-106	0	0	0	0	0	0.410
10	20	I-108	0.129 1,2,4/ 0 3/	0	0	0	0	0
10	20	I-109	0.203	0.203	0	0	0.203	0
10	20	I-110	0.261 1,3,4/ 0.126 2/	0.251 1,3,4/ 0.126 2/	0.261 1,3,4/ 0.126 2/	0.251 1,3,4/ 0.126 2/	0.261 1,3,4/ 0.126 2/	0.251 1,3/ 0.126 2/ 0 4/

TABLE II

(Continued)

m	n	Problem Type & No.	Criterion Used in Phase 2					
			A	B	C	D	E	S
10	20	I-111	0	0	0	0	0	0[1,2] 0.532[3,4]
10	20	I-112	0	0	0	0	0	0.311
10	20	I-113	0.060	0.060	0[1] 0.060[2,3,4]	0.060	0.060	0.341
10	20	I-114	0[1,3] 0.417[2,4]	0[1,3] 0.417[2,4]	0[1,3] 0.417[2,4]	0[1,3] 0.417[2,4]	0[1,3] 0.417[2,4]	0.417[1] 0[2,3,4]
10	20	I-116	0	0	0.705	0[1,3,4] 0.340[2]	0.370	0
10	20	I-117	0	0	0	0	0	0[1,2,3] 0.005[4]
10	20	I-118	0	0	0	0.112[1,2,4] 0[3]	0	0.112

TABLE II

(Continued)

m	n	Problem Type & No.	Criterion Used in Phase 2					
			A	B	C	D	E	S
10	20	I-119	0 1,2,4/ 0.379 3/	0.379 1,2,4/ 0 3/	0.379	0	0	0
10	20	I-120	0 1,2,4/ 1.134 3/	0	0	0 1,2,4/ 1.134 3/	0	0
10	20	Ia-2	0.283	0	0.283	0	0	0
10	20	Ia-3	0 1,2,4/ 0.800 3/	0 1,2,4/ 0.800 3/	0 1,2,4/ 0.800 3/	0 1,2,4/ 0.800 3/	0 1,2,4/ 0.800 3/	0.184 1,2,4/ 0.308 3/
10	20	Ia-4	0 1,2,4/ 0.223 3/	0.272	0.272	0.203	0.272	0.272
10	20	Ia-7	0	0.515	0	0	0	0
10	20	Ia-9	0	0	0	0	0	0.224
10	20	Ic-2	0	0	0	0	0.016	0
10	20	Ic-3	0.054	0.054	0.054	0.054	0	0
15	15	IBM-5	0	0.258	0	0.258	0	0.258

TABLE II

(Continued)

m	n	Problem Type & No.	Criterion Used in Phase 2: A	B	C	D	E	S
12	50	IBM-7	**	**	0	**	0.208	0.153[1,2] 0.059[3,4]
12	37	IBM-8	5.000[1] 6.000[2,4] 7.000[3]	5.000[1,2,4] 0[3]	1.000[1,2,4] 5.000[3]	5.000	1.000[1,2,4] 0[3]	2.000[1,2,4] 7.000[3]
50	15	IBM-9	0	0	0	0	0	0[1,3] 0.258[2,4]
30	60	I-9	0.465[1,2,4] 0.572[3]	0.511[1,2,4] 0.350[3]	0.308[1,2,4] 0.350[3]	0[1,2] 0.350[3,4]	0.308[1,2,4] 0.350[3]	0.330[1] 0.472[2,4] 0.350[3]
Average Without IBM-7		1-2-1	0.242	0.264	0.134	0.214	0.096	0.202
		1-3-1	0.288	0.283	0.146	0.237	0.106	0.197
		2-2-1	0.404	0.100	0.309	0.290	0.090	0.396
		2-3-1	0.293	0.285	0.151	0.240	0.111	0.209

TABLE II

(Continued)

	Problem Type & No.	Criterion Used in Phase 2					
		A	B	C	D	E	S
Average Without IBM-7 & IBM-8	1-2-1	0.066	0.089	0.102	0.037	0.063	0.135
	1-3-1	0.077	0.108	0.114	0.061	0.073	0.130
	2-2-1	0.160	0.104	0.135	0.116	0.093	0.151
	2-3-1	0.082	0.110	0.120	0.064	0.078	0.143

* The listing of just a single number means that all four procedures obtained this same result.

1. Value obtained for 1-2-1.

2. Value obtained for 1-3-1.

3. Value obtained for 2-2-1.

4. Value obtained for 2-3-1.

** No feasible solution obtained in Phase 2.

TABLE III

AVERAGE OF PERFORMANCE DATA FROM 10 × 20 PROBLEMS FOR EACH PHASE 2 CRITERION

Characteristic	Prob. Type	Average						Average Increment		
		1-2A-1	1-2B-1	1-2C-1	1-2D-1	1-2E-1	1-2S-1	1-3-1	2-2-1	2-3-1
Value of α at which a feasible solution is found.	I	0.150	0.143	0.130	0.165	0.148	0.103	-0.024	-0.064	-0.084
	Ia	0.100	0.090	0.065	0.150	0.065	0.095	-0.014	-0.043	-0.057
	Ic	0.090	0.095	0.065	0.135	0.095	0.065	-0.006	-0.061	-0.068
	All	0.122	0.118	0.098	0.154	0.112	0.089	-0.017	-0.058	-0.073
Number of points tried until a feasible solution is found.	I	3.00	2.85	2.60	3.30	2.95	2.15	-0.408	-1.300	-0.308
	Ia	2.00	1.80	1.30	3.00	1.30	1.90	+2.933	-0.867	+3.167
	Ic	1.80	1.90	1.30	2.70	1.90	1.30	-1.333	-1.117	-1.150
	All	2.45	2.35	1.95	3.08	2.28	1.88	+0.196	-1.146	+0.350
Number of trial solutions until a feasible solution is found.	I	4.75	5.70	3.95	6.00	4.30	5.45	-0.308	-1.492	+0.233
	Ia	3.90	4.20	3.13	5.20	3.10	6.30	+5.195	-1.655	+5.728
	Ic	4.60	8.00	5.20	6.20	7.50	6.50	-2.533	-2.800	-2.183
	All	4.50	5.90	4.06	5.85	4.80	5.92	+0.512	-1.860	+0.770
Normalized deviation of feasible solution from $\underline{x}^{(1)}$.	I	1.051	1.102	1.169	1.024	1.086	1.164	-0.020	+0.079	+0.023
	Ia	1.465	1.376	1.409	1.324	1.376	1.481	-0.033	+0.059	-0.018
	Ic	0.958	0.972	0.958	0.948	0.908	0.980	0	+0.007	-0.002
	All	1.131	1.138	1.176	1.080	1.089	1.197	-0.018	+0.056	+0.006

TABLE III
(Continued)

Characteristic	Prob. Type	Average						Average Increment		
		1-2A-1	1-2B-1	1-2C-1	1-2D-1	1-2E-1	1-2S-1	1-3-1	2-2-1	2-3-1
Normalized improvement, first time in Part II of Phase 3.	I	0.077	0.061	0.101	0.061	0.087	0.091	-0.009	+0.025	-0.006
	Ia	0	0	0	0	0	0.036	0	0	0
	Ic	0.075	0.103	0.075	0.090	0.054	0.099	0	+0.006	-0.002
	All	0.057	0.056	0.069	0.053	0.057	0.079	-0.004	+0.008	-0.004
Normalized improvement since first time in Part II of Phase 3.	I	0.093	0.142	0.131	0.092	0.098	0.107	-0.023	+0.036	+0.014
	Ia	0.140	0	0.057	0.005	0.052	0.080	-0.027	-0.013	-0.012
	Ic	0.034	0.021	0.034	0.009	0.009	0.038	0	+0.001	0
	All	0.090	0.076	0.088	0.050	0.064	0.083	-0.018	+0.015	+0.004
Normalized deviation of final solution from $\underline{x}^{(1)}$.	I	0.881	0.898	0.937	0.870	0.901	0.966	+0.012	+0.018	+0.015
	Ia	1.325	1.376	1.352	1.319	1.324	1.365	-0.006	+0.072	-0.006
	Ic	0.849	0.849	0.849	0.849	0.845	0.843	0	0	0
	All	0.984	1.005	1.019	0.977	0.993	1.035	+0.004	+0.027	+0.006
Time for Phase 2 (sec.)	I	0.132	0.132	0.110	0.195	0.195	0.075	-0.014	-0.063	-0.023
	Ia	0.123	0.100	0.069	0.147	0.075	0.066	+0.104	-0.035	+0.140
	Ic	0.069	0.102	0.079	0.100	0.112	0.049	-0.038	-0.037	-0.030
	All	0.114	0.116	0.092	0.159	0.144	0.066	+0.010	-0.050	+0.016
Total time for procedure (sec.)	I	0.367	0.367	0.364	0.491	0.438	0.306	-0.034	-0.053	-0.040
	Ia	0.265	0.205	0.166	0.245	0.147	0.165	+0.089	-0.044	+0.123
	Ic	0.268	0.281	0.295	0.302	0.281	0.246	-0.046	-0.037	-0.032
	All	0.317	0.310	0.297	0.382	0.326	0.256	-0.006	-0.047	+0.003

between $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$ that are used to launch a Phase 2 search until this feasible solution is found. The number of "trial solutions" refers to solutions that are generated during these Phase 2 searches (Step 9) as possible feasible solutions, excluding rounded solutions from the line segment. Normalized deviation is defined just as above, with x_o^B now replaced by the objective function value for $\underline{x}^{(1)}$. The two modes of search used in Method 1 of Phase 3 involve changing just one variable at a time (Part II) and then changing pairs of variables simultaneously. Thus, "Normalized improvement, first time in Part II of Phase 3" shows the decrease in normalized deviation from $\underline{x}^{(1)}$ that would result if only the first mode of search were used to improve upon the original feasible solution, whereas the next set of rows shows the additional improvement from using the full Phase 3. "Total time for procedure" consists of the total CPU time (in seconds) for the entire heuristic procedure <u>except</u> the time required by the simplex method to obtain $\underline{x}^{(1)}$ (which may be several times as large as for the heuristic procedure). No special provisions were made for controlling timing variability on the computer, so the times obtained for individual problems may have a substantial variance due solely to this factor, although the corresponding variance for the average times over 40 problems recorded in Table III would be very much smaller.

6.2. Comparison of Basic Types of Procedures

Tables II, III, IX, and X (see the appendix) provide the primary basis for comparing the four basic types of procedures, 1-2-1, 1-3-1, 2-2-1, and 2-3-1. These rather extensive results fail to reveal any significant differences in the effectiveness of 1-2-1, 1-3-1, and 2-3-1

as measured by the normalized deviation from the best solution. However, they do show 2-2-1 rather consistently lagging behind the other three on the average (see, in particular, the averages given at the bottom of Tables II, IX and X). Furthermore, the time comparison in Table III indicates that 2-2-1 has far too slight an advantage in efficiency to be able to compensate in this way for these substantial differences in effectiveness.

Some of the details in Table III may provide an explanation for the relatively weak performance of 2-2-1. Note that the average "Number of points tried until a feasible solution is found" for 2-2-1 is only about $\frac{1}{2}$ or less that for the other procedures. On the other hand, specific comparison with 2-3-1 on this statistic and on the average "Value of α at which a feasible solution is found" suggests that 2-2-1 has moved somewhat further than the other procedures along the line segment from $\underline{x}^{(1)}$ to $\underline{x}^{(2)}$ before finding a feasible solution. (The α values cannot be directly compared between 1-x-x and 2-x-x procedures since the $\underline{x}^{(2)}$ from Method 2 of Phase 1 tends to be considerably further away from $\underline{x}^{(1)}$ than for Method 1.) Thus, 2-2-1 evidently skips over many rounded solutions near this line segment that would be considered by the other procedures. The result was that the feasible solution obtained in Phase 2 was considerably inferior to those for the other procedures on the average (see the data in the fourth set of rows), and Phase 3 was not fully able to recoup this deficit.

This suggests that the quality of the final solution is affected significantly by the proximity to $\underline{x}^{(1)}$ when initiating the Phase 2 search that leads eventually to this solution at the culmination of Phase 3. Therefore, when moving from $\underline{x}^{(1)}$ toward $\underline{x}^{(2)}$ in Phase 2, it is important to try not to skip over points that may successfully lead to a feasible solution. In other words, it appears that a x-3-x procedure should tend

to be at least slightly more effective on the average than the corresponding x-2-x procedure. (It also tends to require a little more time on the average.)

Although this difference in effectiveness did not become apparent in the comparison between 1-2-1 and 1-3-1, a plausible explanation can again be found in Table III. Specifically, the second set of rows suggest that the jumps along the line segment being made in Phase 2 of 1-2-1 were small enough (except on the Ia problems) to avoid skipping over very many points that were being tried by 1-3-1. However, there is no reason to expect comparable jump sizes in general, particularly since the number of distinct rounded solutions generated by the line segment between $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$ tends to increase with the number of variables in the problem.

Because x-3-x procedures do have considerable variability in the number of points tried and, thus, in the total time for Phase 2, it might prove worthwhile on very large problems to use a compromise between an x-3-x and an x-2-x procedure. In other words, one could begin by selecting points on the line segment between $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$ according to the x-3-x procedure, but after a certain number of futile tries, one could then switch over to using x-2-x. (More complicated ways of merging the two approaches also could be devised.)

Comparing 1-x-x and 2-x-x procedures is inherently very difficult since differences can arise on a given problem only when $\alpha > 0$, which frequently does not occur, and the differences are not likely to be major ones (on the average) unless α is quite large, which seldom happens. The numerous test problems run here certainly provide little basis for choosing between 1-3-1 and 2-3-1. However, there are some clues in Table III that may be significant. In particular, the second, third, and fourth

sets of rows indicate that, for every type of problem, 1-3-1 needed to try less points and less trial solutions in order to obtain a better feasible solution (on the average) than 2-3-1. This suggests that Method 1 of Phase 1 may give a more centralized route into the interior of the feasible region that more readily leads to a good feasible solution than does Method 2. Additional analysis by Faaland and Hillier [2] also suggests that this is the case. Therefore, in the absence of contravening evidence, it is recommended that 1-3-x be adopted as the preferred choice from among the four basic types of procedures tested.

It also should be noted that Faaland and Hillier [3] have analyzed the present Phase 1 methods from a statistical viewpoint. This analysis led them to propose some promising new modifications and extensions of these methods, including the use of a piecewise linear path between $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$. The results of some limited comparative testing also are presented.

6.3. Comparison of Phase 2 Criteria

The same four tables (II, III, IX, and X), plus the additional data and statistical analysis summarized in Table XI of the appendix, provide the primary basis for comparing the six Phase 2 criteria being considered here.

Criteria A and B are the only two that do not consider the objective function, since they are based entirely on (different) measures of infeasibility. Comparing these two criteria first, the tables show Criterion B often lagging substantially behind Criterion A (particularly with the preferred 1-3-1 procedure). Since these criteria differ only in their measure of infeasibility, it appears that A's measure probably is a more effective one than B's.

Criteria C and D use the same measure of improvement p, which does take the objective function into account in a certain way. Otherwise, Criterion C is identical to A, and Criterion D is identical to B. Therefore, to be consistent with the above conclusion on A and B, Criterion C probably should be preferred to D. (The data do not reveal this difference clearly, since Criterion D performs very strongly in Tables II and III, whereas Criterion C does a little better in Tables IX and X.)

The appendix describes how 200 additional test problems were used to try to distinguish between the four remaining criteria - A, C, E, and S. However, even this amount of testing was unable to detect differences at a reasonable level of statistical significance. The main conclusion seems to be that, even though large differences can occur on individual problems, the choice of the criterion does not have a strong effect on the average performance of the heuristic procedure in the long run. All of the available evidence does suggest that the new Criterion S may be somewhat inferior to the others, but that the new Criterion E may be at least as good as any of the others.

Table III indicates that Criterion S does achieve its objective of substantially reducing the time for finding a feasible solution. However, since Phase 3 tends to require somewhat more time than Phase 2, the proportional reduction in the total time for the heuristic procedure is relatively modest. This should tend to be the case for much larger problems as well, except when the "Number of points tried until a feasible solution is found" becomes large, which sometimes would occur with a x-3-x type of procedure.

Table III also reveals two other interesting contrasts between Criterion S and the others. First, with its short and undiscriminating moves toward feasibility, Criterion S tends to more quickly find a feasible

solution in the sense of doing so with a smaller value of α, and so with fewer points tried along the line segment from $\underline{x}^{(1)}$ to $\underline{x}^{(2)}$. At the same time, the resulting feasible solution tended to be inferior to those obtained by the other criteria. Therefore, if the prime objective is to obtain the best possible feasible solution, then it seems better to be more patient and discriminating, as with the other criteria. (The last two sets of rows in Table III indicate that the other criteria average requiring almost twice as much time in Phase 2 as Criterion S, but that the proportional difference in the resulting total time for the overall procedure is considerably less.) Second, the other criteria have a considerably smaller ratio of "number of trial solutions" to "number of points tried" than Criterion S, so they tend to get blocked from making further moves toward feasibility fairly quickly and readily. Therefore, on problems where it is relatively difficult to find any feasible solution, it appears that Criterion S may be more effective in actually reaching such a solution.

7. Evaluation of Phase 3 Methods

7.1. Test Results

The next step in the experimental program was to test the new methods for Phase 3 described in Section 4. Using a fixed Phase 1 and 2 (Methods 1 and 2A, respectively), these three methods plus the old Method 1 were applied to the 18 old problems of Types I and II identified above, the nine IBM problems, and (for economy reasons) just the first four new problems of each of Types I, Ia, and Ic. The resulting normalized deviation of the final solution from the optimal solution is shown in Table IV, along with the total time used in Phase 3. When the optimal solution was not known, a lower bound on the normalized deviation from optimality is shown instead, preceded by a $\geq$ sign. (Since the three phases are not independent, note that

TABLE IV

COMPARISON OF PHASE 3 METHODS

m	n	Problem Type & No.	Normalized Deviation from Optimality				Total Time for Phase 3			
			1-2A-1	1-2A-3	1-2A-4	1-2A-5	1-2A-1	1-2A-3	1-2A-4	1-2A-5
15	15	I-2	0.184	0.184	0.184	0.036	0.07	0.53	0.72	2.20
15	15	I-3	0.169	0.169	0.169	0.076	0.14	0.58	0.67	2.27
15	15	I-4	0.285	0.192	0.192	0.285	0.13	0.69	1.57	1.63
15	15	I-5	0.229	0.229	0.229	0.175	0.11	0.64	0.66	1.60
15	15	I-6	0.279	0.266	0.018	0.109	0.15	0.11	1.00	1.50
15	15	I-7	0.044	0.044	0.044	0.044	0.32	0.11	0.84	1.91
15	15	I-8	0.170	0.170	0.170	0	0.16	0.60	0.78	3.82
15	15	II-1	0.170	0.056	0.056	0	0.17	0.23	1.39	2.24
15	15	II-3	0.032	0	0.032	0.032	0.32	0.13	0.85	1.69
15	15	II-4	0.013	0.013	0.013	0.013	0.15	0.81	0.65	1.24
15	15	II-7	0.018	0.018	0.018	0.018	0.23	0.66	0.93	1.36
15	15	II-9	0.036	0.036	0.036	0.036	0.31	0.33	0.71	1.45
30	15	II-11	0.012	0.012	0.012	0	0.25	2.15	1.12	3.47
15	30	II-13	0.131	0.124	0.131	0.069	1.34	0.33	3.46	6.06
15	30	II-14	0.110	0.104	0.110	0	0.99	2.19	3.19	4.51
30	30	II-15	≥ 0.006	≥ 0.064	≥ 0.006	≥ 0.006	2.12	0.41	5.59	10.40
30	30	II-16	≥ 0.017	≥ 0.035	≥ 0.017	≥ 0.017	1.94	0.62	5.80	9.62
10	20	I-101	0.587	0.282	0.174	0.174	0.31	1.31	0.92	1.52
10	20	I-102	≥ 0.052	≥ 0.052	≥ 0.052	≥ 0.015	0.15	0.66	0.87	1.92
10	20	I-103	≥ 0.280	≥ 0.280	≥ 0.280	≥ 0	0.19	0.15	0.90	1.51
10	20	I-104	≥ 0.208	≥ 0.208	≥ 0.208	≥ 0.063	0.23	0.74	0.98	2.55
10	20	Ia-1	0.456	0.101	0.101	0	0.07	0.15	0.63	1.17
10	20	Ia-2	1.162	0.053	0.053	0.088	0.26	0.18	0.74	5.37
10	20	Ia-3	1.280	0	0	0	0.03	0.30	0.33	1.17
10	20	Ia-4	0.622	0.272	0.272	0.272	0.10	0.37	0.41	1.29
10	20	Ic-1	0.356	0.078	0.078	0.116	0.19	1.00	1.16	0.77
10	20	Ic-2	0.913	0	0	0	0.19	0.07	0.74	1.23
10	20	Ic-3	1.077	0.185	0.185	0.238	0.14	0.20	0.80	0.18

TABLE IV

(Continued)

m	n	Problem Type & No.	Normalized Deviation from Optimality				Total Time for Phase 3			
			1-2A-1	1-2A-3	1-2A-4	1-2A-5	1-2A-1	1-2A-3	1-2A-4	1-2A-5
10	20	Ic-4	≥ 0.155	≥ 0.036	≥ 0.036	≥ 0	0.14	0.51	1.25	0.74
7	7	IBM-1	0.378	0	0	0	0.04	0.09	0.11	0.44
7	7	IBM-2	0	0	0	0	0.02	0.03	0.05	0.34
3	4	IBM-3	0	0	0	0	0.03	0	0.05	0.02
15	15	IBM-4	0.258	0.258	0.258	0.258	0.14	0.39	0.81	1.66
15	15	IBM-5	0.258	0.258	0.258	0.258	0.08	0.47	0.88	1.55
31	31	IBM-6	0.360	0.180	0.180	0.180	0.38	0.60	0.81	5.61
12	50	IBM-7*	0	0.075	0	0	0.56	1.85	2.27	3.36
12	37	IBM-8	13	13	0	0	0.05	1.11	0.79	0.13
50	15	IBM-9	0	0	0	0	0.15	0.84	0.91	4.03
30	60	I-9	≥ 0.465	≥ 0.465	≥ 0.465	≥ 0.459	2.01	1.86	20.91	23.33
Average			≥ 0.610	≥ 0.449	≥ 0.104	≥ 0.078	0.37	0.62	1.69	3.00
Average without IBM-8			≥ 0.283	≥ 0.119	≥ 0.107	≥ 0.080	0.38	0.61	1.71	3.08

*Criterion C was used on this problem since Criterion A did not give a feasible solution in Phase 2.

the relative performance of the Phase 3 methods could change somewhat if a different combination of methods for Phases 1 and 2 were used; budget limitations prevented checking this further.)

7.2. Comparison of Phase 3 Methods

The last comment of Sec. 6.3 actually has a bearing on comparing Phase 3 methods also. Method 5 follows a philosophy similar to Criterion S for Phase 2 in that it is relatively undiscriminating in the individual moves it allows in order to expedite making combinations of moves that may indeed lead to good feasible solutions. By contrast, Methods 1, 3, and 4 are analogous to the other criteria in that they set high requirements for each individual move, and so tend to get blocked from making further moves toward improved feasible solutions fairly quickly and readily. The results of Table IV suggest that the less discriminating approach of Method 5 tends to be more effective in actually reaching improved feasible solutions. However, the much longer (albeit fewer) searches involved consume considerable time, and Table IV indicates that the increase in total time over the other methods tends to be substantial. In particular, Method 5 has an Average Total Time for Phase 3 in Table IV that is approximately 2, 5 and 8 times as large as that for Methods 4, 3 and 1, respectively.

Method 3 emerges as a "best buy" approach in terms of the trade-off between execution time and the quality of the solution attained.

The growth rates of total time with problem size in Table IV appear to be roughly comparable for the different methods. However, there may be small differences in the growth rates, so it is uncertain whether the time comparisons between methods observed here would still hold for very large problems.

Finally, note that the fifth and sixth set of rows in Table III show that going on to the second mode of search in Method 1 tends to be very

worthwhile, more than doubling the improvement from just the first mode of search alone (except for Ic problems) on the average. However, the fact that Method 3 performed nearly as well as Method 4 in Table IV demonstrates that little is lost by foregoing the second mode of search if the new mode introduced by these methods also is being used. In fact, the second mode of search achieved a larger improvement than the new mode used in Method 3 (or 4) on only 3 of the 39 problems, and was outperformed 17 times. Furthermore, Method 4 (which uses both the second mode and the new mode) achieved a further improvement over the second mode on 15 of the 35 problems where optimality had not yet been reached. Nevertheless, comparing Methods 4 and 5 in Table IV shows that this new mode itself was outperformed by the new mode used in Method 5 on 14 of the 19 problems where differences occurred. The Method 5 mode of search actually achieved a further improvement over the second mode 25 out of the 35 possible times.

The unescapable conclusion is that changing only one or two variables is frequently inadequate for reaching a better feasible solution. It may be necessary to change many variables. Since it wouldn't be computationally feasible to investigate such simultaneous changes directly, one needs a mode of search that accomplishes this indirectly by making a long sequence of small changes. Some of these small changes may need to worsen the situation, when considered individually, in order to permit combinations of changes which provide a total overall improvement, a la Method 5.

8. Evaluation of Multiple Solution Procedures

8.1. Test Results

The investigation next considered the question of the best way to generate multiple solutions in order to try to improve upon the first final solution. The first such method tested was the one proposed in [9] of repeating the Phase 2 search over the entire line segment between $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$ in order to generate a series of feasible solutions as starting points for Phase 3. The results of doing this with Method 3 of Phase 2 (which provides the most exhaustive possible search over this line segment) are shown in Table V for Procedures 1-R3-1 and 2-R3-1 under all six Phase 2 criteria.

Procedure x-R3x-1 is guaranteed to do at least as well as the corresponding x-2x-1 (or x-3x-1) procedure (since it obtains the same solution as well as others), and it sometimes will provide a significant improvement. In fact, for the problems in Table V where the x-2x-1 procedure did not obtain an optimal solution, x-R3x-1 gave an improvement on essentially half of them, and the average fractional improvement (when it did occur) on the original normalized deviation from optimality was approximately $\frac{1}{2}$.

Table VI provides a more detailed analysis of this approach for the 1-R3A-1 procedure. The number of distinct solutions in Phase 1 refers to the number of rounded solutions that were obtained from the line segment between $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$. Each of these new rounded solutions initiates a Phase 2 search for a feasible solution, and the "Ph.2" column gives the number of times this search actually obtained a new feasible solution. Since each such solution leads to a final solution from Phase 3, the "Ph.3" column shows how many distinct final solutions were thereby obtained. The "First Solution" column gives the range of α that would successfully lead to the first final solution obtained. The "Best Solution" column

TABLE V

NORMALIZED DEVIATION FROM OPTIMALITY

FOR THE x-R3x-1 PROCEDURES

			Phase 1 Method						Phase 2 Criterion					
m	n	Problem Type & No.	1,A	1,B	1,C	1,D	1,E	1,S	2,A	2,B	2,C	2,D	2,E	2,S
15	15	I-2	0	0	0	0	0	0	0.036	0.036	0.036	0.036	0.036	0.036
15	15	I-3	0.169	0.169	0.169	0.169	0.169	0.169	0.169	0.169	0.169	0.169	0.169	0.169
15	15	I-4	0.279	0.279	0.279	0.279	0.279	0.279	0.279	0.279	0.279	0.279	0.279	0.279
15	15	I-5	0.229	0.229	0.229	0.229	0.229	0.229	0.229	0.229	0.229	0.229	0.229	0.229
15	15	I-6	0	0	0	0	0	0	0.164	0.164	0.164	0.164	0.164	0.018
15	15	I-7	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044
15	15	I-8	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044	0.044
15	15	II-1	0.170	0.170	0.170	0.170	0.170	0.170	0.170	0.170	0.170	0.170	0.170	0.170
15	15	II-3	0	0	0	0	0	0	0	0	0	0	0	0
15	15	II-4	0.013	0.013	0.013	0.013	0.013	0	0	0	0	0	0	0
15	15	II-7	0.018	0.018	0.018	0.018	0.018	0.018	0.018	0.018	0.018	0.018	0.018	0.018
15	15	II-9	0.036	0.036	0.036	0.036	0.036	0.036	0	0	0	0	0	0
30	15	II-11	0.012	0.012	0.012	0.012	0.012	0.012	0.012	0.012	0.012	0.012	0.012	0.012
15	30	II-13	0.072	0.072	0.072	0.072	0.072	0.072	0.131	0.131	0.131	0.131	0.131	0.131
15	30	II-14	0	0	0	0	0	0	0	0	0	0	0	0
30	30	II-15	$\geq$0.006	$\geq$0.006	$\geq$0.006	$\geq$0.006	$\geq$0.006	$\geq$0.030	$\geq$0.006	$\geq$0.006	$\geq$0.006	$\geq$0.006	$\geq$0.006	$\geq$0.006
30	30	II-16	$\geq$0.017	$\geq$0.017	$\geq$0.017	$\geq$0.017	$\geq$0.017	$\geq$0.017	$\geq$0.017	$\geq$0.012	$\geq$0.046	$\geq$ 0	$\geq$0.017	$\geq$0.046
10	20	I-101	0.565	0.413	0.565	0.565	0.565	0.565	0.565	0.413	0.565	0.565	0.565	0.565

TABLE V

(Continued)

m	n	Problem Type & No.	Phase 1 Method, 1,A	1,B	1,C	1,D	1,E	1,S	Phase 2 Criterion, 2,A	2,B	2,C	2,D	2,E	2,S
10	20	I-102	≥0.052	≥0.052	≥0.052	≥ 0	≥0.052	≥0.052	≥ 0	≥ 0	≥ 0	≥ 0	≥ 0	≥ 0
10	20	I-103	≥ 0	≥ 0	≥ 0	≥ 0	≥ 0	≥0.280	≥ 0	≥ 0	≥ 0	≥ 0	≥ 0	≥0.280
10	20	I-104	≥ 0	≥ 0	≥ 0	≥ 0	≥ 0	≥ 0	≥0.009	≥0.009	≥ 0	≥ 0	≥0.009	≥0.009
10	20	Ia-1	0.368	0.368	0.368	0.368	0.368	0.368	0.368	0.368	0.368	0.368	0.368	0.368
10	20	Ia-2	0.879	0.879	0.879	0.879	0.879	0.879	0.879	0.879	0.879	0.879	0.879	0.879
10	20	Ia-3	1.280	1.280	1.280	1.280	1.280	1.465	1.280	1.280	1.280	1.280	1.280	1.465
10	20	Ia-4	0.350	0.350	0.437	0.437	0.350	0.437	0.350	0.350	0.437	0.437	0.350	0.437
10	20	Ic-1	0.356	0.356	0.356	0.356	0.356	0.356	0.356	0.356	0.356	0.356	0.356	0.356
10	20	Ic-2	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913	0.913
10	20	Ic-3	1.023	1.023	1.023	1.023	1.023	1.023	0.838	0.838	0.838	0.838	0.838	0.838
10	20	Ic-4	≥0.155	≥0.155	≥0.155	≥0.155	≥0.155	≥0.155	≥0.140	≥0.140	≥0.140	≥0.140	≥0.140	≥0.140
7	7	IBM-1	0.378	0.378	0.378	0.378	0.378	0.378	0.378	0.378	0.378	0.378	0.378	0.378
7	7	IBM-2	0	0	0	0	0	0	0	0	0	0	0	0
3	4	IBM-3	0	0	0	0	0	0	0	0	0	0	0	0
15	15	IBM-4	0.258	0.258	0.258	0.258	0.258	0.258	0.258	0.258	0.258	0.258	0.258	0.258
15	15	IBM-5	0.258	0.516	0.258	0.516	0.258	0.516	0.258	0.516	0.258	0.516	0.258	0.516
31	31	IBM-6	0.180	0.180	0.180	0.180	0.180	0.180	0.180	0.180	0.180	0.180	0.180	0.180

TABLE V

(Continued)

m	n	Problem Type & No.	1,A	1,B	1,C	1,D	1,E	1,S	2,A	2,B	2,C	2,D	2,E	2,S
			Phase 1 Method						Phase 2 Criterion					
12	50	IBM-7	*	*	0	*	0.208	0.059	*	*	0	*	0.208	0.059
12	37	IBM-8	12	8	9	11	8	10	12	8	8	8	8	10
50	15	IBM-9	0	0	0	0	0	0	0	0	0	0	0	0
30	60	I-9	≥ 0	≥ 0	≥ 0	≥ 0	≥ 0	≥0.330	≥0.350	≥0.350	≥0.308	≥0.350	≥0.308	≥0.350
Average without IBM-7			≥0.530	≥0.427	≥0.453	≥0.511	≥0.424	≥0.507	≥0.538	≥0.435	≥0.434	≥0.441	≥0.432	≥0.504
Average without IBM-7 & IBM-8			>0.194	>0.222	>0.222	>0.228	>0.220	>0.251	>0.202	>0.231	>0.230	>0.237	>0.227	>0.247

*No feasible solution obtained in Phase 2.

TABLE VI

SUMMARY OF PERFORMANCE FOR PROCEDURE 1-R3A-1

m	n	Problem Type & No.	No. of Distinct Solutions			First Solution	Best Solution		Decr. in Norm. Dev. from Opt.
			Ph.1	Ph.2	Ph.3	α	α	Ph.2 Sol$^{\underline{n}}$	
15	15	I-2	11	7	7	0 -0.269	0.782-0.801	#4	0.184
15	15	I-3	4	3	3	0 -0.572	0 -0.572 0.646-1	#1 #3	0
15	15	I-4	13	10	6	0 -0.081	0.082-0.389 0.668-0.828	#2-3 #6-7	0.006
15	15	I-5	14	12	3	0.129-0.870	0.129-0.870	#1-8	0
15	15	I-6	6	5	2	0 -0.751	0.752-1	#4-5	0.279
15	15	I-7	9	7	5	0 -0.153	0 -0.153	#1	0
15	15	I-8	11	9	6	0 -0.050	0.942-1	#9	0.126
15	15	II-1	2	2	1	0 -1	0 -1	#1-2	0
15	15	II-3	3	2	2	0 -0.391	0.392-1	#2	0.032
15	15	II-4	3	2	1	0 -1	0 -1	#1-2	0
15	15	II-7	3	2	2	0 -0.660	0 -0.660	#1	0
15	15	II-9	1	1	1	0 -1	0 -1	#1	0
30	15	II-11	4	3	1	0 -1	0 -1	#1-3	0
15	30	II-13	2	2	2	0 -0.208	0.209-1	#2	0.059
15	30	II-14	4	4	4	0 -0.302	0.585-0.594	#3	0.110
30	30	II-15	5	4	4	0 -0.135	0 -0.135	#1	0
30	30	II-16	9	9	8	0 -0.081	0 -0.081 0.350-0.402	#1 #6	0
10	20	I-101	7	5	5	0 -0.287	0.288-0.327	#2	0.022
10	20	I-102	8	6	4	0.236-0.586	0.236-0.586	#1-2	0
10	20	I-103	16	13	9	0.131-0.256	0.257-0.310	#2	0.280
10	20	I-104	9	8	6	0.352-0.405	0.799-0.801 0.865-1	#5 #8	0.208

TABLE VI

(Continued)

m	n	Problem Type & No.	No. of Distinct Solutions Ph.1	Ph.2	Ph.3	First Solution α	Best Solution α	Ph.2 Sol$^{\underline{n}}$	Decr. in Norm. Dev. from Opt.
10	20	Ia-1	6	5	3	0 -0.132	0.563-1	#4-5	0.088
10	20	Ia-2	12	9	3	0 -0.158	0.159-0.499	#2-3	0.283
10	20	Ia-3**	77	42	34	0.190-0.217	0.190-0.217	#1	0
10	20	Ia-4**	50	37	24	0 -0.023	0.024-0.055	#2	0.272
10	20	Ic-1	6	5	1	0 -1	0 -1	#1-5	0
10	20	Ic-2	9	6	2	0 -0.952	0 -0.952	#1-5	0
10	20	Ic-3	8	4	2	0 -0.574	0.575-1	#2-4	0.054
10	20	Ic-4	4	1	1	0.576-1	0.576-1	#1	0
7	7	IBM-1	1	1	1	0 -1	0 -1	#1	0
7	7	IBM-2	2	2	1	0 -1	0 -1	#1-2	0
3	4	IBM-3	1	1	1	0 -1	0 -1	#1	0
15	15	IBM-4	1	1	1	0 -1	0 -1	#1	0
15	15	IBM-5	2	2	2	0 -0.750	0 -0.750	#1	0
31	31	IBM-6	7	7	4	0 -0.062	0.063-0.679	#2-5	0.180
12	50	IBM-7	6	0	0	-	-	-	-
12	37	IBM-8	22	13	3	0.061-0.068	0.069-0.200	#2	1.000
50	15	IBM-9	10	7	6	0 -0.418	0 -0.418	#1-2	0
30	60	I-9	28	24	19	0.057-0.058	0.184-0.185 0.221-0.284	#3 #5	0.465
Average			10.2	7.3	4.9	0.046-0.512	0.209-0.698*	#2.1-2.9*	0.097

*When more than one interval for α yielded the best solution, the widest interval was used for purposes of calculating the average.

**Linear extrapolation was used to estimate "No. of distinct solutions" since the run was terminated before $\alpha = 1$.

shows the range of α that would lead to the best final solution obtained, and then identifies which of the distinct feasible solutions obtained in Phase 2 (in chronological order of increasing α) led to this best solution. The final column shows the resulting decrease in the normalized deviation from optimality compared to the 1-2A-1 procedure.

After making two observations, Table VI also indicates how the computational effort of 1-R3A-1 on these problems compares with that for 1-3A-1 (or 1-2A-1). First, the time required for Phase 1 (other than obtaining $x^{(1)}$ by the simplex method) is negligible compared to Phases 2 and 3. Second, the time required to go through Phases 2 and 3 again to generate a new (not necessarily distinct) final solution tends to be about the same as for obtaining the initial final solution with 1-3A-1 (or 1-2A-1). Therefore, the "No. of Distinct Solutions in Ph. 2" column indicates approximately the multiple of the time for 1-3A-1 (excluding the simplex method) that was required for 1-R3A-1.

Table VII summarizes the average performance of the x-R3x-1 procedures on these same 39 problems for all 12 combinations of Phase 1 methods and Phase 2 criteria. In addition to showing the same type of data as in Table VI (where "Decr. in Norm. Dev. from Opt." now is in comparison with the corresponding x-2x-1 procedure), the last two sets of rows contrasts the results being obtained at those times when the procedure currently is initiating the Phase 2 search from the two extreme points $(\alpha = 0, 1)$ and from the equivalent of the midpoint $(\alpha = \frac{1}{2})$ of the line segment between $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$. The average times given for these three cases include just the time required for Phase 2 to find a feasible solution and for Phase 3 to then obtain a final solution.

TABLE VII

AVERAGE OF PERFORMANCE DATA FOR EACH x-R3x-1 PROCEDURE

Characteristic		Phase 1 Method						Phase 2 Criterion					
		1,A	1,B	1,C	1,D	1,E	1,S	2,A	2,B	2,C	2,D	2,E	2,S
Number of Distinct Solutions	Ph.1	10.2	10.2	10.2	10.2	10.2	10.2	26.4	26.4	26.4	26.4	26.4	26.4
	Ph.2	7.3	7.3	7.5	7.3	7.1	7.9	19.0	19.9	19.9	19.8	19.5	20.6
	Ph.3	4.9	5.0	4.8	4.6	4.9	5.1	9.8	10.7	10.4	10.3	10.5	10.9
First Solution	α	0.046 ↓ 0.512	0.093 ↓ 0.538	0.046 ↓ 0.513	0.123 ↓ 0.533	0.046 ↓ 0.524	0.033 ↓ 0.447	0.016 ↓ 0.297	0.037 ↓ 0.312	0.016 ↓ 0.298	0.045 ↓ 0.321	0.016 ↓ 0.296	0.016 ↓ 0.284
Best Solution	α	0.209 ↓ 0.698	0.250 ↓ 0.715	0.232 ↓ 0.704	0.278 ↓ 0.725	0.206 ↓ 0.703	0.214 ↓ 0.693	0.153 ↓ 0.459	0.170 ↓ 0.504	0.147 ↓ 0.486	0.166 ↓ 0.498	0.152 ↓ 0.486	0.146 ↓ 0.492
	Ph.2 Sol.	#2.1 ↓ 2.9	#2.1 ↓ 2.9	#2.4 ↓ 3.3	#2.1 ↓ 2.9	#1.9 ↓ 2.8	#2.4 ↓ 3.3	#2.8 ↓ 6.0	#2.9 ↓ 6.2	#2.9 ↓ 6.2	#2.9 ↓ 6.1	#2.7 ↓ 5.9	#3.1 ↓ 6.5
Decr. in Norm. Dev. from Opt.		0.097	0.206	0.066	0.107	0.084	0.059	0.097	0.196	0.082	0.182	0.077	0.068
Norm. Dev. from Opt.*	α=0	0.169	0.169	0.186	0.172	0.169	0.173	0.169	0.169	0.186	0.172	0.169	0.173
	α=0.5	0.154	0.154	0.150	0.154	0.190	0.190	0.186	0.186	0.183	0.183	0.186	0.182
	α=1.0	0.181	0.181	0.181	0.181	0.181	0.181	0.249	0.249	0.249	0.249	0.249	0.269
Time* (sec.)	α=0	0.46	0.48	0.43	0.45	0.43	0.41	0.46	0.48	0.44	0.45	0.44	0.38
	α=0.5	0.43	0.45	0.43	0.45	0.40	0.40	0.54	0.57	0.55	0.54	0.54	0.51
	α=1.0	0.56	0.57	0.56	0.59	0.55	0.55	0.63	0.63	0.62	0.64	0.60	0.64

*These averages are only over those 19 problems where a feasible solution was obtained at $\alpha = 0$ for all 12 cases.

The next method tested for generating multiple solutions was the Rx-x-x procedure presented in Section 2. Table VIII shows the results of applying the R1-2A-1 version of this new type of procedure to the same 39 problems. Thus, Solution 0 refers to the ordinary final solution obtained by the 1-2A-1 procedure, whereas Solution i $(i = 1, \ldots, 5)$ refers to the solution generated by applying this same procedure with the $i^{\underline{th}}$ new $\underline{x}^{(1)}$ and $\underline{x}^{(2)}$ obtained as described in Section 2. The next-to-last column shows the resulting improvement over Solution 0 by taking the best of the six solutions. Comparing this column with the last column of Table VI thereby provides one comparison of the Rx-x-x and x-Rx-x types of procedures. However, it must be noted that the specific procedure reported in Table VI is 1-R3A-1, where the number of final solutions generated is highly variable and possibly very large. Therefore, a better comparison might be to the 1-R2A-1 procedure with $\alpha = 0, 0.2, 0.4, 0.6, 0.8, 1.0$, where six final solutions (at most) also would be generated. The last column of Table VIII shows the improvement over the ordinary 1-2A-1 procedure given by this 1-R2A-1 procedure.

The reason for constructing the final column of Table VIII in this way is that it essentially equates the computational effort for 1-R2A-1 to that required by R1-2A-1 to obtain the results in the next-to-last column. After excluding the time required to solve for $x^{(1)}$ by the simplex method, the time required to obtain each new final solution tends to be about the same whether it is done by 1-2A-1, R1-2A-1, or 1-R2A-1. Therefore, the time required for each of the last two columns is approximately six times that for 1-2A-1.

TABLE VIII

NORMALIZED DEVIATION FROM OPTIMALITY

FOR THE SOLUTIONS GENERATED BY PROCEDURE R1-2A-1

m	n	Problem Type & No.	Solution 0	1	2	3	4	5	Improvement Over 1-2A-1: R1-2A-1	1-R2A-1
15	15	I-2	0.184	1.036	2.136	1.966	0.823	0.106	0.078	0.184
15	15	I-3	0.169	0.134	0.216	0.274	0.280	0.216	0.035	0
15	15	I-4	0.285	0.279	0.682	0.440	0.484	1.160	0.006	0.006
15	15	I-5	0.229	1.044	0.845	0.960	0.229	0.175	0.054	0
15	15	I-6	0.279	0.321	0.303	0.394	0.466	0.527	0	0.279
15	15	I-7	0.044	0.255	1.603	0.532	0.372	0.292	0	0
15	15	I-8	0.170	0.239	0.069	0.516	0.044	0.258	0.126	0.126
15	15	II-1	0.170	0.309	0.086	0.163	0.219	0.219	0.084	0
15	15	II-3	0.032	0.084	0.261	0.197	0.177	0.205	0	0.032
15	15	II-4	0.013	0.134	0.013	0.081	0	0.291	0.013	0
15	15	II-7	0.018	0.046	0.230	0.051	0.018	0.175	0	0
15	15	II-9	0.036	0.123	0.059	0	0.319	0	0.036	0
30	15	II-11	0.012	0.172	0	0.316	0	0.025	0.012	0
15	30	II-13	0.131	0.093	0.117	0.131	0	0.131	0.131	0.059
15	30	II-14	0.110	0.110	0.129	0.110	0.110	0.270	0	0.034
30	30	II-15	≥0.006	≥0.006	≥0.006	≥0.006	≥0.006	≥0.006	0	0
30	30	II-16	≥0.017	≥0.029	≥0.032	≥0.029	≥0.104	≥0.017	0	0
10	20	I-101	0.587	0.152	0.261	1.102	0.858	0.803	0.435	0
10	20	I-102	≥0.052	≥0	≥0.892	≥1.804	≥0.984	≥0.722	0.052	0
10	20	I-103	≥0.280	≥9.849	≥16.870	≥8.829	≥12.343	*	0	0
10	20	I-104	≥0.208	≥1.111	≥4.021	≥2.924	≥4.173	≥1.026	0	0.208
10	20	Ia-1	0.456	0.658	0	0.253	0	0.101	0.456	0.088
10	20	Ia-2	1.162	0.601	1.343	0.884	0.601	0.601	0.561	0.283
10	20	Ia-3	1.280	4.305	11.807	7.010	1.968	16.542	0	0
10	20	Ia-4	0.622	0.631	0.408	0.359	0.594	0.173	0.449	0
10	20	Ic-1	0.356	0.181	0.350	0.214	0.201	0.214	0.175	0
10	20	Ic-2	0.913	0.288	0.445	0	0	0.722	0.913	0

TABLE VIII

(Continued)

m	n	Problem Type & No.	Solution 0	1	2	3	4	5	Improvement Over 1-2A-1: R1-2A-1	1-R2A-1
10	20	Ic-3	1.077	0.083	0.408	0.238	0.520	0.418	0.994	0.054
10	20	Ic-4	$\geq$0.155	$\geq$0.181	$\geq$0.129	$\geq$0.155	$\geq$0.150	$\geq$1.185	0.026	0
7	7	IBM-1	0.378	0	0	0	0	0	0.378	0
7	7	IBM-2	0	0	0	0	0	0	0	0
3	4	IBM-3	0	0	0	0	0	0	0	0
15	15	IBM-4	0.258	0.258	0.258	0.258	0.258	0.258	0	0
15	15	IBM-5	0.258	0.258	0.258	0.258	0.258	0.258	0	0
31	31	IBM-6	0.360	0.180	0.180	0.180	0.180	0.180	0.180	0.180
12	50	IBM-7	*	*	*	*	*	*	*	*
12	37	IBM-8	13	15	13	12	13	15	1.000	0
50	15	IBM-9	0	1.033	0.775	0.516	0.775	0	0	0
30	60	I-9	$\geq$0.465	$\geq$0.272	$\geq$0.578	$\geq$0.468	$\geq$0.597	$\geq$0.819	0.193	0.078
Average without IBM-7			$\geq$0.625	$\geq$1.038	$\geq$1.547	$\geq$1.148	$\geq$1.082	-	0.168	0.042
Average without IBM-7 & I-103			$\geq$0.635	$\geq$0.800	$\geq$1.132	$\geq$0.940	$\geq$0.778	$\geq$1.165	0.173	0.044
Average without IBM-7, I-103 & IBM-8			$\geq$0.291	$\geq$0.406	$\geq$0.803	$\geq$0.633	$\geq$0.438	$\geq$0.780	0.150	0.045
Average without IBM-7, IBM-8, I-103, I-104, Ia-3			$\geq$0.264	$\geq$0.270	$\geq$0.384	$\geq$0.378	$\geq$0.283	$\geq$0.310	0.159	0.048

*No feasible solution obtained in Phase 2.

8.2. Comparison of Multiple Solution Procedures

Three basic approaches now are available for generating multiple solutions at one phase in order to try to improve upon the final solution obtained at the end of Phase 3. One is the x-Rx-x approach, which is investigated in detail in Tables V, VI, and VII for the x-R3x-1 case. A second is the Rx-x-x approach, which is studied in Table VIII for the case of R1-2A-1. The third is to apply different individual procedures to the same problem, e.g., by varying the criterion used in Phase 2.

One could increase the number of solutions obtained even further by combining all three approaches. For example, consider all of the procedures represented in Tables IV-VIII as applied to the same 39 problems. On 9 of these problems, the same solution was obtained by essentially every procedure. However, there was considerable variation in the solutions obtained, and in the relative performances on the individual procedures, on the other 30 problems. Specifically, the best solution was obtained by the 1-2A-5 procedure 13 times, by the R1-2A-1 procedure 12 times, by some x-R3x-1 procedure 12 times, by the 1-2A-4 procedure 7 times, and by the 1-2A-3 procedure 7 times. The average normalized deviation from optimality of the best solution obtained on each of the 39 problems was only ≥ 0.045, a very substantial improvement over that obtained by any one of the procedures. Furthermore, this best solution actually was optimal on 19 of the 32 problems where the optimal solution was known. (Keep in mind that many of these problems had been selected as "difficult" ones for which previously tested heuristic procedures had not obtained an optimal solution.)

These facts illustrate a key inherent property of heuristic procedures, namely, the great variability in their relative performances from one

problem to the next. Even though one procedure may be distinctly better than another on the average, the inferior procedure may still obtain a better solution on a considerable proportion of the problems. Given the highly combinatorial nature of integer programming problems, there is a good deal of luck involved in uncovering a good (or optimal) feasible solution, and sometimes a less promising path will lead to the prize. Therefore, if the only concern were to maximize the quality of the best solution obtained, then one should indeed follow the old maxim to "try, try again."

However, the time and expense involved also must be considered, of course. One can quickly reach a point of diminishing return where the expected marginal improvement from additional runs becomes too small to justify the computer time. This is particularly true if the new solutions being obtained frequently are merely repeats of ones previously obtained. Therefore, an important criterion for designing a multiple-solution procedure is its tendency to generate new distinct solutions.

Under this criterion, the approach of repeating the same individual procedure with different Phase 2 criteria, as in Table II, is not a particularly good one. (Also see Table V.) Even varying the Phase 1 and 2 methods (from among the choices represented in Table II) does not help much. (Note that all 24 procedures only obtained about $2\frac{1}{2}$ distinct solutions per problem in Table II.) Varying the Phase 3 method, as in Table IV, does somewhat better. Although Methods 3 and 4 usually give the same solution, this solution frequently is different from the Method 5 solution, so running either x-x-3 or x-x-4 along with x-x-5 would be reasonable. Nevertheless, 1-2A-3 and 1-2A-4 each were able to improve upon 1-2A-5 only five times (six times combined) in Table IV, and the improvements obtained were quite modest.

The x-Rx-x approach fares somewhat better under this criterion. In fact, Table VI shows that 1-R3A-1 obtained an average of 4.9 distinct solutions (out of 10.2 attempts) per problem. Of the 36 problems where the first solution (the 1-3A-1 solution) was not optimal, a subsequent solution provided an improvement on 17 of them. Similarly, Table VIII indicates that 1-R2A-1 provided an improvement over 1-2A-1 on 13 of these problems.

However, Table VIII demonstrates that the Rx-x-x approach is the easy champion for generating distinct solutions. In fact, with the exception of the IBM problems, practically every solution is distinct. Furthermore, R1-2A-1 improved over 1-2A-1 on 23 of the problems, with an average improvement far larger than either 1-R2A-1 or 1-R3A-1. Even when compared with the champion among the tested individual procedures, 1-2A-5, R1-2A-1 obtained a better solution on 9 of the problems, and R1-2A-5 probably would have done much better.

The apparent explanation for the great success of Rx-x-x in generating distinct solutions is the diversity of its areas of search. Whereas x-Rx-x continues searching along the same line segment into the feasible region, Rx-x-x uses a completely different line segment in a different part of the feasible region to search for each new solution.

Since Rx-x-x requires only about the same computer time per solution (and so less per distinct solution) as the corresponding x-Rx-x procedure, it must be rated as the better of these multiple-solution approaches. The test results on individual procedures can guide the choice of the specific Rx-x-x procedure, e.g., R1-3E-5 should be a very powerful one.

However, the choice between x-Rx-x and Rx-x-x need not be an either-or one. For a fixed total number of solutions to be generated, some can be obtained in one way and the rest in the other. Since the computer time

involved for obtaining each solution is about the same for comparable x-Rx-x and Rx-x-x procedures, this combined approach would be advantageous if the expected marginal improvement from another solution sometimes would be larger if it is generated by x-Rx-x rather than Rx-x-x. There is some evidence suggesting that this is indeed the case.

To begin examining this evidence, consider the "Ph.2" and "Ph.2 Sol^n" columns of Table VI. Although there is considerable variation in which Phase 2 solution led to the best final solution, a careful examination suggests a definite bias toward the earlier (smaller α) Phase 2 solutions (particularly when the total number of such solutions is relatively large). Compare the averages. Then note that #1 led to the best solution 21 out of 38 times, and #2 did so 19 out of 33 possible times, whereas #4 did so only 8 out of 24 possible times. Of 21 problems with five or more distinct Phase 2 solutions, only two required going beyond #4 to obtain the best solution. Ten problems required going to #2 to obtain the best solution (contributing 0.058 of the 0.097 average decrease in the normalized deviation from optimality), and five more required going to either #3 or #4. Thus, going to #2 is very worthwhile, but the expected marginal improvement decreases rapidly thereafter.

An analysis of Table VIII suggests that Rx-x-x gives a slower decrease in the expected marginal improvement of its successive solutions than x-Rx-x. Solution 4 obtained the best solution as many times (16) as Solution 1. Ten problems required going to Solution 1 to obtain the best solution, but 13 more required going further (4 to Solution 2, and 3 each to Solutions 3, 4, and 5). The average marginal improvement was 0.097 for Solution 1, 0.023 for Solution 2, and a total of 0.048 for the next three solutions.

However, after going to at least Solution 1, the first extra solution from 1-R3A-1 (#2) gave a larger average marginal improvement than the next R1-3A-1 solution. After going to at least Solution 3, the next 1-R3A-1 solution (#3) helped more than the next R1-3A-1 solution.

Another factor arguing for a combined approach is that Rx-x-x is not well-suited for all problems. Specifically, if the extreme points adjacent to $\underline{x}^{(1)}$ are located far away from the good feasible solutions (because their objective function values are far inferior to that for $x^{(1)}$), then Rx-x-x will have little chance of helping. This apparently occurred on problems I-103, I-104, and Ia-3 in Table VIII, since each of the adjacent extreme points on these problems led to final solutions with extremely large normalized deviations from optimality.

Therefore, the recommended multiple-solution approach is to combine Rx-x-x and x-Rx-x (denoted hereafter as Rx-Rx-x). For a specified total number of solutions to be generated, the first solution would be obtained in the usual way, then the desired number of x-Rx-x solutions, and then Rx-x-x would be applied for the rest. (This idea could be pushed even further by using the x-Rx-x approach to generate more than one solution along each of the <u>new</u> line segments generated by the Rx-x-x approach, but this would not seem worthwhile under ordinary circumstances.) To be specific on the allocation of effort, it is suggested that 1/3 of the solutions (rounded down) be generated by x-Rx-x.

Among the alternative Rx-Rx-x procedures, R1-R3E-5 should be a very effective one. With x-x-5 rather than x-x-1, the expected marginal improvement of successive solutions should decrease even more rapidly, so the number generated should be kept rather small. Four should be very adequate for most purposes, and it rarely should be worthwhile to exceed ten or fifteen.

9. Comparison with the Ibaraki-Ohashi-Mine Algorithm

The algorithm presented by Ibaraki et al [10] is (with some modifications) a 2-R3-x procedure, where their Phase 3 method is perhaps most similar to Method 5 of the ones tested here. For part of their testing program, they obtained a listing of all of the author's test problems used in [9] (except the five "Large Test Problems"), as given in Appendix 1 of [8]. They report the results of running their algorithm on five of these problems, namely, I-5, I-6, II-1, II-3, and II-11, achieving a normalized deviation from optimality of 0.229, 0.018, 0.047, 0, and 0.012, respectively, for an average of 0.061.

Comparing these numbers with those given in Tables IV, V, and VII for these problems suggests that this algorithm is quite competitive with 1-2A-5, x-R3x-1, or R1-2A-1. However, since it is most similar to 2-R3x-5, the most interesting comparisons would be with this procedure and its recommended variations, 1-R3X-5 and R1-R3x-5. Although these particular data are not available, consider the approximation where 1-2A-5 is used for the first solution and 1-R3x-1 (rather than the more powerful 1-R3x-5) for the rest. This approach obtains a better solution than the Ibaraki algorithm on four of the five problems and matches the optimal solution on the fifth. (When 1-2A-5 is combined with 2-R3A-1 or R1-R2A-1 (with few solutions) instead, then the one change is that the Ibaraki algorithm does better on I-6.)

It must be emphasized, however, that more data would be required to draw a conclusive comparison.

10. Some Hypothesized Principles for Heuristic Algorithms

Some of the insights gained in this investigation may have wider applicability for the development of heuristic algorithms in various areas of discrete programming, etc. They can at least provide promising starting points for guiding the development of such algorithms and the resulting experience would determine the extent to which they are generally valid. Therefore, they are presented here in the form of hypothesized principles for heuristic algorithms. **At the same time, it must be emphasized that these are only tentative hypotheses, based on a limited set of test problems which may not be representative of many other discrete programming problems.**

Heuristic can be defined as "guiding in discovery" or, in the present context, "guiding in the discovery of good solutions." No guarantees can be given on how successful the search actually will be, and results may vary considerably from one attempt to the next. There is a definite element of luck involved, where the heuristic procedure is designed to weight the odds in one's favor as much as possible. Therefore, it seems appropriate to use the parlance of gambling in stating these hypothesized principles. (Translations follow in parentheses.)

Hypothesized Principle 1: Keep your stakes high. (Focus the search where any feasible solutions found should be particularly good ones, even though they are difficult to uncover, rather than quickly locating a mediocre feasible solution and trying to improve upon it.)

The heuristic procedures studied herein start the search from the "ideal," the linear programming solution $\underline{x}^{(1)}$, so any feasible solutions nearby will be particularly good ones (as well as particularly difficult to uncover). The focus of the search moves no further away from this ideal than is necessary to find the initial feasible solution for Phase 3.

Early computational experience showed that this approach was far superior to the alternative of merely rounding $\underline{x}^{(2)}$ to quickly obtain a (mediocre) initial feasible solution, and also was markedly superior to such intermediate alternatives as Method 1 of Phase 2. As discussed in Section 6, even the slight compromises involved in Method 2 or Criterion S of Phase 2 seems to have a detrimental effect. Thus, there is a rather strong positive correlation between the quality of the initial feasible solution and of the final solution obtained.

Hypothesized Principle 2: Stick with a winning streak, but don't push your luck. (When a good feasible solution is found, there often is an even better one nearby, so continue searching in this region until no further improvement have been obtained within a reasonable period of time.)

The first two modes of search used in most of the Phase 3 methods seek (and often find) small changes in the current feasible solution that yield improvements. However, care is taken not to use a mode of search whose growth rate with problem size is so large that it can continue fruitlessly for an extended time. On a broader basis, the x-Rx-x multiple-solution approach also initiates new searches near where the last successful one began. As discussed in Section 6, this approach is a good one if it is used in moderation.

Hypothesized Principle 3: Occasionally shuffle the deck. (When the search has gotten too locked into one well explored neighborhood, then allow it to drift awhile until it homes in on better solutions in an entirely different part of the feasible region.)

Method 5 of Phase 3 examplifies the successful application of this principle.

Hypothesized Principle 4: Play the field. (The search should move into several different "high stake" regions in order to obtain a variety of solutions, thereby increasing the chances of "hitting the jackpot" of finding an optimal solution.)

The Rx-x-x multiple-solution approach enjoyed outstanding success in following this principle.

11. Summary of Recommendations and Conclusions

Although an extensive program of computational experimentation failed to reveal significant differences between certain of the leading methods and criteria, the following heuristic procedures can be recommended from the current evidence as at least comparable to any of the others on the problems tested.

1. For a basic single-solution procedure that is both very effective and reasonably efficient, use 1-3E-5.
2. For a very powerful multiple-solution procedure, use R1-R3E-5 as described at the end of Section 8.2. The total number of solutions generated should be adjusted to fit the desired time-quality tradeoff, but four should be very adequate for most purposes.
3. For a "quick and dirty" procedure that is still reasonably effective, use 1-2S-3. (The procedure can even be terminated early as needed and still probably provide a reasonably good feasible solution since it quickly finds an initial feasible solution and then devotes the bulk of the time to seeking progressive improvements.)

No further comparative testing is planned. However, a modest amount of additional developmental testing to refine these procedures appears to be worthwhile. The primary focus will be on Method 5 of Phase 3 to see if it can be streamlined greatly without significantly diminishing its effectiveness, but several similar questions about other parts of the procedures also will be pursued. Further information on the growth rate of execution time with problem size also will be sought.

In a parallel investigation, Faaland and Hillier [3] have analyzed the current Phase 1 methods in detail and developed some new options.

Although the procedures tested here have been presented in terms of pure integer programming, they can be extended quite readily to the more general case of mixed integer programming. Ibaraki et al [10] have done this in one way. This also has been done in another way by the author, and some preliminary testing conducted. These results will be presented subsequently in conjunction with the further testing and refining mentioned above.

In many problems, the integer variables actually are restricted to be binary (0-1). These procedures could, of course, be applied directly to such problems by merely introducing upper bounds of one on the variables as additional functional constraints. However, this would be a crude and needlessly inefficient way of handling 0-1 problems. What is needed is a thorough adaptation and streamlining of the procedures and codes to exploit the special structure of these problems. This is being pursued.

REFERENCES

1. Balas, E., and C. Martin, "Pivot and Complement - A Heuristic for 0-1 Programming," Management Science, Vol. 26 (1980), pp. 86-96.

2. Echols, R. E., and L. Cooper, "Solution of Integer Linear Programming Problems by direct Search," Journal of the Association for Computing Machinery, Vol. 15 (1968), pp. 45-84.

3. Faaland, Bruce H., and Frederick S. Hillier, "Interior Path Methods for Heuristic Integer Programming Procedures," Operations Research, Vol. 27 (1979), pp. 1069-1087.

4. Faaland, Bruce H., and Frederick S. Hillier, "The Accelerated Bound-and-Scan Algorithm for Integer Programming," Operations Research, Vol. 23 (1975), pp. 406-425.

5. Glover, F., "Heuristics for Integer Programming Using Surrogate Constraints," Decision Sciences, Vol. 8 (1977), pp. 156-166.

6. Haldi, John, "25 Integer Programming Test Problems," Working Paper No. 23, Graduate School of Business, Stanford University, Dec., 1964 (multilith).

7. Hillier, Frederick S., "A Bound-and-Scan Algorithm for Integer Linear Programming," Operations Research, Vol. 17 (1969), pp. 638-679.

8. Hillier, Frederick S., "A Bound-and-Scan Algorithm for Pure Integer Linear Programming with General Variables," Appendix 1 (Listing of Test Problems), Technical Report No. 11 (ONR Contract Nonr-225(89)), Department of Operations Research, Stanford University, 1969.

9. Hillier, Frederick S., "Efficient Heuristic Procedures for Integer Linear Programming with An Interior," Operations Research, Vol. 17 (1969), pp. 600-637.

10. Ibaraki, Toshihide, Tateaki Ohashi, and Hisashi Mine, "A Heuristic Algorithm for Mixed-Integer Programming Problems," Mathematical Programming Study 2, (1974), pp. 115-136.

11. Jacqmin, Nancy Eileen, "Documentation of a Computer Program for Hillier's Heuristic Procedure in Integer Linear Programming," Technical Report No. 88 (ONR Contract N00014-75-C-0418) and 79-8 (NSF Grant MCS76-81259 A01), Department of Operations Research, Stanford University, 1979.

12. Jeroslow, R., and T. Smith, "Experimental Results on Hillier's Linear Search," Mathematical Programming, Vol. 9 (1975), pp. 371-376.

13. Kochenberger, G. A., B. A. McCarl, and F. P. Wyman, "A Heuristic for General Integer Programming," Decision Sciences, Vol. 5 (1974), pp. 36-44.

14. Reiter, S., and D. B. Rice, "Discrete Optimizing Solution Procedures for Linear and Nonlinear Integer Programming Problems," Management Science, Vol. 12 (1966), pp. 829-850.

15. Roth, R. H., "An Approach to Solving Linear Discrete Optimization Problems," Journal of the Association for Computing Machinery, Vol. 17 (1970), pp. 300-313.

16. Senju, S., and Y. Toyoda, "An Approach to Linear Programming with 0-1 Variables," Management Science, Vol. 15 (1968), pp. B196-B207.

17. Toyoda, Y., "A Simplified Algorithm for Obtaining Approximate Solutions to Zero-One Programming Problems," Management Science, Vol. 21 (1975), pp. 1417-1427.

18. Trauth, C. A., Jr., and R. E. Woolsey, "Integer Linear Programming: A Study in Computational Efficiency," Management Science, Vol. 15 (1969), pp. 481-493.

19. Zanakis, Stelios H., "Heuristic 0-1 Linear Programming: An Experimental Comparison of Three Methods," Management Science, Vol. 24 (1977-78), pp. 91-104.

APPENDIX

Further Comparative Testing

In order to more adequately compare the four basic types of procedures and the six Phase 2 criteria considered in Table II, a supplemental experimental program was undertaken.

The first step was to randomly generate 20 more Type I problems (labeled I-121 to I-140) and 20 more Type Ia problems (labeled I-21 to I-40), each with $m = 10$ and $n = 20$. (These two types were chosen since, based on the resulting normalized deviations from optimality, they appear to be particularly difficult for the heuristic procedures.) All of the procedures and criteria were applied to these problems, with the results shown in Tables IX and X, respectively.

Given the analysis of computational results presented in the early part of Section 6, the next step was to randomly generate 200 additional problems of Type I (again with $m = 10$, $n = 20$) to try to distinguish between Criteria A, C, E, and S on a sound statistical basis. As the recommended choice at this point of the four basic types of procedures, procedure 1-3-1 was used with each of these criteria. On 29 of the problems, the simplex method found that there was either no feasible solutions or no bounded optimal solution $\underline{x}^{(1)}$ for the continuous (linear programming) version of the problem. Each of the four criteria produced a feasible integer solution on all of the remaining 171 problems. The six pairs of criteria were then compared on the quality of their final solutions on these 171 problems.

Let $V_i(x)$ denote the normalized deviation from optimality of the final solution under Criterion x ($x = A, C, E, S$) on problem i ($i = 1, 2, \ldots, 171$). For a given pair of criteria, x and y, let

$$D_i(x\text{-}y) = V_i(x) - V_i(y), \quad \text{for} \quad i = 1, 2, \ldots, 171 .$$

TABLE IX

NORMALIZED DEVIATION FROM BEST SOLUTION
FOR NEW TYPE I PROBLEMS

Problem	Criterion Used in Phase 2					
	A	B	C	D	E	S
I-121	0	0	0	0	0	0[1,3] 0.430[2] 0.107[4]
I-122	0.232	0.232	0.232	0.232	0.232	0
I-123	0[1,4] 0.251[3] 0.178[2]	0[1,4] 0.488[3] 0.178[2]	0[1,4] 0.488[3] 0.178[2]	0[1,4] 0.488[3] 0.178[2]	0[1,4] 0.251[3] 0.178[2]	0.251[1,2] 0.488[3] 0[4]
I-124	0	0	0	0	0	0
I-125	0[1,2] 0.144[3] 0.313[4]	0[1,2] 0.144[3] 0.726[4]	0[1,2,4] 0.144[3]	0[1,2] 0.144[3] 0.313[4]	0[1,2,4] 0.144[3]	.159[1] 0[2], .144[3] 0.313[4]
I-126	0	0	0.360	0	0	0
I-127	0.032	0.032	0.032	0.032	0.032	0[1] 0.032[2,3,4]
I-128	0	0	0	0	0	.233
I-129	0[1,2,4] 0.481[3]	0[1,2,4] 0.739[3]	0[1,2,4] 0.481[3]	0	0[1,2,4] 0.481[3]	0.481
I-130	3.209[1,3] 0[2,4]	3.209[1,3] 1.502[2] 2.144[4]	0.642[1,2] 3.209[3] 1.635[4]	0.642[1,2] 3.209[3,4]	3.209[1,3] 1.502[2] 2.14[illegible][4]	3.209[1,3] 0[2,4]

TABLE IX

(Continued)

Problem	Criterion Used in Phase 2					
	A	B	C	D	E	S
I-131	0.189[1] 0.023[2,3,4]	0.189	0	0.189[1] 0.023[2,3,4]	0	0
I-132	0[1,2,4] 0.037[3]	0[1,2,4] 0.037[3]	0[1,2] 0.037[3,4]	0[1,2] 0.037[3,4]	0[1,2] 0.037[3,4]	0
I-133	0.399[1,3,4] 0[2]	0.399	0	0	0	0.399
I-134	0	0	0	0	0	0
I-135	0.379	0.379	0.379	0.379	0.379	0[1,2] 0.379[3,4]
I-136	0	0	0	0	0	0[1,3,4] 0.428[2]
I-137	0.099	0.099	0.283	0.099[1] 0.499[2] 0.283[3,4]	0.099[1,3] 0.499[2,4]	0
I-138	0.540	0.223[1,2,4] 0.758[3]	0.115	0.223[1,2,4] 0.758[3]	0.115	0
I-139	0	0	0	0	0	0.361
I-140	0	0.116	0	0.021	0.021	0

TABLE IX

(Continued)

Problem		Criterion Used in Phase 2					
		A	B	C	D	E	S
Ave.	1-2-1	0.254	0.244	0.102	0.091	0.204	0.255
	1-3-1	0.074	0.167	0.111	0.131	0.148	0.131
	2-2-1	0.291	0.341	0.288	0.280	0.235	0.286
	2-3-1	0.101	0.227	0.154	0.238	0.171	0.115
Ave. w/o I-130	1-2-1	0.098	0.088	0.074	0.062	0.046	0.099
	1-3-1	0.078	0.097	0.083	0.105	0.077	0.138
	2-2-1	0.138	0.190	0.134	0.126	0.078	0.132
	2-3-1	0.107	0.126	0.076	0.081	0.068	0.121

1. Value obtained for 1-2-1.
2. Value obtained for 1-3-1.
3. Value obtained for 2-2-1.
4. Value obtained for 2-3-1.

TABLE X

NORMALIZED DEVIATION FROM BEST SOLUTION
FOR NEW TYPE Ia PROBLEMS

Problem	Criterion Used in Phase 2					
	A	B	C	D	E	S
Ia-21	0	0	0	0	0	0
Ia-22	0	0	0	0	0	0
Ia-23	0.139[1] 0[2,3,4]	0.139[1] 0[2,3,4]	0.139[1,2] 0.500[3] 0[4]	0.139[1] 0[2,4] 0.500[3]	0.139[1,2] 0.500[3] 0[4]	0.139[1,2] 0[3,4]
Ia-24	0	0	0	0	0	0
Ia-25	0[1,2,4] 1.869[3]	0[1,2,4] 1.869[3]	0[1,2,4] 1.869[3]	0[1,2,4] 1.869[3]	0[1,2,4] 1.869[3]	0[1,2,4] 1.869[3]
Ia-26	0[1,2,4] 0.176[3]	0[1,2,4] 0.176[3]	0.026[1,4] 0[2] 0.176[3]	0.026[1,4] 0[2] 0.176[3]	0.026[1,4] 0[2] 0.176[3]	0.026[1,4] 0[2] 0.176[3]
Ia-27	0[1,2,4] 0.635[3]	0[1,2,4] 0.635[3]	0[1,2,4] 0.635[3]	0[1,2,4] 0.635[3]	0[1,2,4] 0.635[3]	0[1,2,4] 0.635[3]
Ia-28	0	0	0	0	0	0
Ia-29	0	0.479	0[1,2,4] 0.479[3]	0.479	0[1,2,4] 0.479[3]	0
Ia-30	0	0	0	0	0	0
Ia-31	0	0	0	0	0	0
Ia-32	0	0	0	0	0	0

TABLE X

(Continued)

Problem	A	B	C	D	E	S
	Criterion Used in Phase 2					
Ia-33	0	0[1,2] 0.719[3,4]	0[1,2] 0.719[3,4]	0[1,2] 0.719[3,4]	0[1,2] 0.719[3,4]	0
Ia-34	1.549[1,2,4] 0.801[3]	1.549[1,2,4] 0.801[3]	0.801[1,3] 1.549[2] 0[4]	1.549[1,2] 0.801[3] 0[4]	1.549[1,2] 0.801[3,4]	1.549[1,2,4] 0.801[3]
Ia-35	0.514[1,2,4] 0[3]	0[1,3] 0.514[2,4]	0[1,3,4] 0.514[2]	0	0[1,3,4] 0.514[2]	0.514[1,2,4] 0[3]
Ia-36	0	0	0	0	0	0
Ia-37	0[1,2,4] 0.564[3]	0[1,2,4] 0.564[3]	0[1,2,4] 0.564[3]	0[1,2,4] 0.564[3]	0[1,2,4] 0.564[3]	0
Ia-38	0[1,2] 0.667[3] 0.485[4]	0[1,2] 0.667[3] 0.171[4]	0	0[1,3,4] 0.171[2]	0	1.083[1] 0[2],0.667[3] 0.171[4]
Ia-39	0.072[1,2] 0[3,4]	0.072[1,2,4] 0[3]	0.072[1,2,4] 0[3]	0.072[1,2,4] 0[3]	0.072[1,2] 0[3,4]	0.072[1,2] 0[3,4]
Ia-40	0.011	0.011	0.011[1,2] 0[3,4]	0.011[1,2] 0[3,4]	0.011[1,2] 0[3,4]	0.011
Ave. 1-2-1	.114	.113	.052	.114	.090	.170
Ave. 1-3-1	.107	.131	.114	.114	.114	.114
Ave. 2-2-1	.236	.296	.287	.287	.287	.208
Ave. 2-3-1	.128	.176	.041	.065	.077	.114

1. Value obtained for 1-2-1.
2. Value obtained for 1-3-1.
3. Value obtained for 2-2-1.
4. Value obtained for 2-3-1.

Since the problems are randomly generated (by the mixed congruential method) from fixed (discretized uniform) probability distributions, all of these D_i for the given x and y are drawn from the same underlying probability distribution (before conditioning on the parameter values taken on for the particular problem). In order to distinguish between the two criteria involved, the objective is to determine whether the mean of this underlying distribution can be concluded to be different from zero and, if so, whether it is positive (y better than x) or negative (x better than y).

Since any two criteria (x and y) frequently yield the same final solution, this underlying distribution has considerable mass at zero, which makes it considerably more difficult to detect a nonzero mean. However, this mean indeed will be nonzero if the rest of the distribution (the conditional distribution given nonzero values) has a nonzero mean, so it is better to focus on this conditional distribution to address the questions at issue. Let μ_{x-y} denote the mean of this conditional distribution. The nonzero $D_i(x-y)$ values represent random observations from the conditional distribution that can be used to obtain a point estimate of μ_{x-y} and a confidence interval about μ_{x-y}. For this purpose, it is assumed that the conditional distribution is (approximately) normal. (If no assumption is made about the functional form of this distribution, the resulting confidence interval would be much wider, which would further reinforce the conclusions stated after the next paragraph.

Using this approach for each of the six pairs of criteria led to the results shown in Table XI. The first column of data shows the sample size n_{x-y} (the number of nonzero $D_i(x-y)$ values out of the 171 problems). The next two columns give the maximum likelihood estimate of μ_{x-y} (the sample average), denoted by $\hat{\mu}_{x-y}$, and the sample standard deviation s_{x-y}. The last column presents the 99% confidence interval about μ_{x-y} (so that the fiduciary probability that all six intervals cover their respective means is at least 0.94 by the Bonferroni inequality).

TABLE XI

STATISTICAL COMPARISON OF PHASE 2 CRITERIA
FROM 200 ADDITIONAL TYPE I PROBLEMS

x-y	n_{x-y}	$\hat{\mu}_{x-y}$	s_{x-y}	99% Conf. Int.
A-C	41	+0.013	0.391	(-0.152, 0.178)
A-E	42	+0.025	0.414	(-0.148, 0.198)
A-S	69	-0.081	0.736	(-0.316, 0.155)
C-E	27	+0.019	0.342	(-0.164, 0.202)
C-S	73	-0.084	0.704	(-0.302, 0.135)
E-S	74	-0.089	0.706	(-0.307, 0.128)

Recalling that positive values of μ_{x-y} favor y over x, it can be seen that the results in Table XI are most favorable to Criterion E (by a slight margin over C and A), and least favorable to Criterion S. However, it must be emphasized that none of the confidence intervals exclude zero, so the null hypothesis that $\mu_{x-y} = 0$ cannot be rejected in any of the six cases at a 99% level of statistical significance. In fact, even for the most extreme case of E-S, this hypothesis cannot be rejected at just the 90% level, even if the alternative hypothesis is the one-sided one, $\mu_{E-S} < 0$. Also note how very small the $\hat{\mu}_{x-y}$ are relative to the s_{x-y}. Therefore, the only solid conclusion that can be drawn from these data is that, even though there will be occasional large differences in both directions on individual problems, any differences in the long run average performance of these criteria should be quite small on problems similar to Type I.

DATE
ILME